UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/782,422	02/19/2004	Ahmad Said Ghazal	11187 and 11490	6283

7590 08/08/2006	EXAMINER
John D. Cowart	SYED, FARHAN M
Teradata Law IP, WHQ-4W	

ART UNIT	PAPER NUMBER
2165	

NCR Corporation
1700 S. Patterson Blvd.
Dayton, OH 45479-0001

DATE MAILED: 08/08/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

	Application No.	Applicant(s)
Office Action Summary	10/782,422	GHAZAL, AHMAD SAID
	Examiner	Art Unit	
	Farhan M. Syed	2165	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE <u>3</u> MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.
- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *19 February 2004*.

2a)☐ This action is **FINAL**.   2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-56* is/are pending in the application.

    4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-56* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *19 February 2004* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

    Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

    Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

    a)☐ All   b)☐ Some *   c)☐ None of:

      1.☐ Certified copies of the priority documents have been received.

      2.☐ Certified copies of the priority documents have been received in Application No. _____.

      3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

    * See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1)☒ Notice of References Cited (PTO-892)

2)☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3)☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)
    Paper No(s)/Mail Date _____.

4)☐ Interview Summary (PTO-413)
    Paper No(s)/Mail Date. _____ .

5)☐ Notice of Informal Patent Application (PTO-152)

6)☐ Other: _____.

DETAILED ACTION

1.   Claims 1-56 are pending.

Double Patenting

2.   The nonstatutory double patenting rejection is based on a judicially created

doctrine grounded in public policy (a policy reflected in the statute) so as to prevent the

unjustified or improper timewise extension of the "right to exclude" granted by a patent

and to prevent possible harassment by multiple assignees.   A nonstatutory

obviousness-type double patenting rejection is appropriate where the conflicting claims

are not identical, but at least one examined application claim is not patentably distinct

from the reference claim(s) because the examined application claim is either anticipated

by, or would have been obvious over, the reference claim(s). See, e.g., *In re Berg*, 140

F.3d 1428, 46 USPQ2d 1226 (Fed. Cir. 1998); *In re Goodman*, 11 F.3d 1046, 29

USPQ2d 2010 (Fed. Cir. 1993); *In re Longi*, 759 F.2d 887, 225 USPQ 645 (Fed. Cir.

1985); *In re Van Ornum*, 686 F.2d 937, 214 USPQ 761 (CCPA 1982); *In re Vogel*, 422

F.2d 438, 164 USPQ 619 (CCPA 1970); and *In re Thorington*, 418 F.2d 528, 163

USPQ 644 (CCPA 1969).

A timely filed terminal disclaimer in compliance with 37 CFR 1.321(c) or 1.321(d)

may be used to overcome an actual or provisional rejection based on a nonstatutory

double patenting ground provided the conflicting application or patent either is shown to

be commonly owned with this application, or claims an invention made as a result of

activities undertaken within the scope of a joint research agreement.

Effective January 1, 1994, a registered attorney or agent of record may sign a terminal

disclaimer. A terminal disclaimer signed by the assignee must fully comply with 37 CFR

3.73(b).

Claims 1-56 rejected on the ground of nonstatutory obviousness-type double

patenting as being unpatentable over claims 1-17 of U.S. Patent No. 6,990,484 B1.

Although the conflicting claims are not identical, they are not patentably distinct from

each other because the grant of a second patent would give rise to an unjustified

extension of the rights granted in the first patent.

Claim Rejections - 35 USC § 101

3.   35 U.S.C. 101 reads as follows:

> Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

Claims 1-3, 7, 9-12, 14, 15, 17-20, 21-23, 27, 29-31, 34, 35, 37, 38, 40, 45, 47-

49, 52, 53, 55, and 56 are rejected under 35 U.S.C. 101 because the claimed invention

is directed to non-statutory subject matter. The claims recite a determining step that

may also include a mental process, where the mental process does not produce a

tangible result. Any claim that does not produce a tangible result is directed to a non-

statutory subject matter.

Claim Rejections - 35 USC § 102

4.   Claims 1-56 are rejected under 35 U.S.C. 102(e) as being anticipated by Ghazal

et al (U.S. Patent No. 6,990,484 B1 and known hereinafter as Ghazal).

The applied reference has a common assignee with the instant application.

Based upon the earlier effective U.S. filing date of the reference, it constitutes prior art

under 35 U.S.C. 102(e). This rejection under 35 U.S.C. 102(e) might be overcome

either by a showing under 37 CFR 1.132 that any invention disclosed but not claimed in

the reference was derived from the inventor of this application and is thus not the

invention "by another," or by an appropriate showing under 37 CFR 1.131.

As per claims 1, 21, and 39, Ghazal teaches a method for analyzing a query (i.e.

"The method includes analyzing the satisfiability of the conditions before executing the query.")(Column 1,

lines 31-33), the query including one or more conditions and one or more sub-queries (i.e.

"The DBS 100 usually receives queries and commands to build tables in a standard format, such as

SQL.")(Column 5, lines 26-28), the conditions including one or more connecting conditions

that introduce the sub-query in the query, each of the sub-queries including zero or

more conditions, the method including (i.e. *"The conditions include one or more variables, zero or*

more SQL IN (inlist) conditions, and zero or more SQL NOT IN (not in list) conditions.")(Column 3, lines

50-53): determining the satisfiability of the query (i.e. *"Satisfiability of the conditions is analyzed*

before executing the query.")(Abstract), including: determining the satisfiability of the

connecting conditions (i.e. *"If the query optimizer of the database has the ability to check if a set of*

conditions is satisfiable, or "SAT," then such queries could be answered immediately without accessing

the data.")(Column 1, lines 17-19); and determining the satisfiability of the conditions in the sub-queries (i.e. *"If the query optimizer of the database has the ability to check if a set of conditions is satisfiable, or "SAT," then such queries could be answered immediately without accessing the data.")*(Column 1, lines 17-19).

As per claims 2, 22, and 40, Ghazal teaches a method where determining the satisfiability of the query further includes determining the satisfiability of all other conditions (i.e. *"Convert all conditions to <= comparisons only using the following transformation (block 405, expanded in FIG. 5)...")*(Column 7, lines 7-9).

As per claims 3, 12, 23, 32, 41 and 50 Ghazal teaches a method where determining the satisfiability of the conditions includes: creating and populating a global conditions set (i.e. *"The conditions may include one or more variables, one or more SQL IN (inlist) conditions, and one or more SQL NOT IN (not in list) conditions. Analyzing the satisfiability of the conditions may include: for each variable in the query, defining an in list from a SQL IN statement associated with the variable, a list from the SQL NOT IN statement's not in list and any other query statements that relate the variable to a constant with a operator, and an interval for the range of values associated with the variable; finding all components, where each component includes variables related in the query by an equals relation...")*(Column 2, lines 48-62); and determining the satisfiability of the global conditions set (i.e. *"...determining that the conditions are not satisfiable if any of the following conditions are met:...")*(Column 3, lines 4-5).

As per claims 4, 24, and 42, Ghazal teaches a method where the query includes a clause of the form (X CC (SELECT Y FROM T)), where CC is a connecting condition,

X and Y are variables or columns, T is a set of one or more tables or views, and where

populating the global conditions set includes: if CC is "IN," adding (X=Y) to the global

conditions set; if CC is "NOT IN," adding (XGhazalY) to the global conditions set; and if

CC includes arithmetic comparison COMP, adding (X COMP Y) to the global conditions

set (i.e. *"In general, in another aspect, the invention features a method for determining the satisfiability of*

and creating transitive closure in conditions in a database query. The conditions include one or more

variables, zero or more SQL IN (inlist) conditions, and zero or more SQL NOT IN (not in list) conditions.

The method includes for each variable in the query, defining an in list from a SQL IN statement

associated with the variable, a list from the SQL NOT IN statements not in list and any other query

statements that relate the variable to a constant with a operator, and an interval for the range of values

associated with the variable.")(Column 3, lines 48-61).

As per claims 5, 25, and 43, Ghazal teaches a method where the query includes

a clause of the form (CC (SELECT Y FROM T WHERE R)), where CC is a connecting

condition, Y is a variables or a column, T is a set of one or more tables or views, and R

is a set of one or more conditions, and where populating the global conditions set

includes adding R to the global conditions set (i.e. *"In many cases, users would like to query*

orders for a specific month or months. For example, "SELECT * FROM ORDERTBL where

EXTRACT(MONTH FROM O.sub.--ORDERDATE)=9" retrieves all orders for the month of

September.")(Column 6, lines 47-51).

As per claims 6, 16, 26, 36, 44 and 54, Ghazal teaches a method where

determining the satisfiability of the global conditions set includes: converting the form of

the conditions in the global conditions set to less-than-or-equal-to conditions; creating a

map M of the less-than-or-equal-to conditions (i.e. *"Analyzing satisfiability may include*

converting the conditions to less-than-or-equal-to conditions, creating a map M of the less-than-or-equal-

to conditions...")(Column 1, lines 37-40); finding the shortest path between all nodes in M (i.e.

"...finding the shortest path between all nodes in M...")(Column 1, line 39); and determining if M has

a negative cycle and, if it does, returning that the query is not satisfiable (i.e.

"...determining if M has a negative cycle and, if it does, returning that the conditions are not

satisfiable.")(Column 1, lines 41-43).

As per claims 7, 17, 27, 37, 45 and 55, Ghazal teaches a method where creating

the map M of the conditions in the global conditions set includes: creating a node for

each of the variables in the conditions; creating a node for 0 (i.e. *"11. Create a weighted*

directed graph M=[V,E] (block 410). V is the graph's nodes composed of the variables in the constraints

plus a special node for 0. E is the set of edges and it reflects the constraints in the following way (FIG.

6)")(Column 7, lines 29-33); creating a directed edge from a node representing a first

variable, S, to a node representing a second variable, T, with a cost, C, for conditions of

the form (S<=T+C) (i.e. *"A directed edge from X to Y with cost C for (X<=Y+C) (block 605)"*)(Column

7, lines 34-35); creating a directed edge from a node representing a first variable, S, to the

0 node, with cost C, for conditions of the form (S<=0+C) (i.e. *"A directed edge from X to 0 with*

cost C for (X<=0+C) (block 610)")(Column 7, lines 36-37); and creating a directed edge from the

0 node to a node representing a first variable, S, with cost C, for conditions of the form

(0<=X+C) (i.e. *"A directed edge from 0 to X with cost C for (0<=X+C) (block 615)"*)(Column 7, lines 38-

39).

As per claims 8, 28, and 46, Ghazal teaches a method where finding the shortest path between all nodes in M includes running the Floyd-War shall Shortest Path Algorithm against M (i.e. *"12. Find the shortest path between all nodes in M using "Floyd-Warshall Shortest Path Algorithm" (block 415). The resulting updated M also will have the shortest paths between the nodes."*)(Column 7, lines 40-43).

As per claims 9, 29, and 47, Ghazal teaches a method where determining if M has a negative cycle includes determining if M includes a negative cost edge from a node to itself (i.e. *"13. The set of constraints is "contradictory" if and only if (block 420) M has a negative cost edge from a node to itself. If so, return FALSE (block 425)."*)(Column 7, lines 44-46).

As per claims 10, 18, 30, 38, 48 and 56 Ghazal teaches a method where analyzing the query further includes: determining the transitive closure of the conditions and, if necessary, modifying the conditions (i.e. *"If the conditions are satisfiable, the method includes analyzing the transitive closure of the conditions and modifying the conditions to meet transitive closure, if necessary, before executing the query."*)(Column 1, lines 32-36).

As per claims 11, 31, and 49, Ghazal teaches a method where the query includes an outer query block and an inner query block, and where determining the transitive closure of the conditions and modifying the conditions includes: determining the transitive closure of conditions in the outer query block and, if necessary, modifying the conditions in the outer query block and determining the transitive closure of conditions in the inner query block and, if necessary, modifying the conditions in the

inner query block (i.e. *"The conditions may include one or more variables, one or more SQL IN (inlist)*

conditions, and one or more SQL NOT IN (not in list) conditions. Analyzing the satisfiability of the

conditions may include: for each variable in the query, defining an in list from a SQL IN statement

associated with the variable, a list from the SQL NOT IN statement's not in list and any other query

statements that relate the variable to a constant with a operator, and an interval for the range of values

associated with the variable; finding all components, where each component includes variables related in

the query by an equals relation...")(Column 2, lines 48-62).

As per claims 13, 33, and 51, Ghazal teaches a method where the transitive

closure includes one or more transitive closure conditions, and where modifying the

conditions to achieve transitive includes: for each transitive closure condition of the form

(COL COMP C), where COL is a column, COMP is a comparison, and C is a constant: if

COL appears in the outer query block, adding the transitive closure condition to the

outer query block; and if COL appears in the inner query block, adding the transitive

closure condition to the inner query block (i.e. *"In general, in another aspect, the invention*

features a method for determining the satisfiability of and creating transitive closure in conditions in a

database query. The conditions include one or more variables, zero or more SQL IN (inlist) conditions,

and zero or more SQL NOT IN (not in list) conditions. The method includes for each variable in the query,

defining an in list from a SQL IN statement associated with the variable, a list from the SQL NOT IN

statements not in list and any other query statements that relate the variable to a constant with a

operator, and an interval for the range of values associated with the variable.")(Column 3, lines 48-61).

As per claims 14, 34, and 52, Ghazal teaches a method for analyzing a query (i.e.

"The method includes analyzing the satisfiability of the conditions before executing the query.")(Column 1,

lines 31-33), the query including one or more conditions of the form (X+Y OP C), where X and Y are variables, C is a constant, and OP is an operator, the method including (i.e. *"One well known algorithm for determining SAT, investigated by Rosenkrantz and Hunt, fits well in the pertinent scope of conditions. This algorithm considers only conjunctive conditions, where each condition is of the form (X op Y+C) or (X op C). Both X and Y are integer variables and C is an integer constant and op .epsilon.[<,=,>,>=,<=]."*)(Column 7, lines 1-6): determining the satisfiability of the query, including: determining the satisfiability of the one or more conditions of the form (X+Y OP C) (i.e. *"The function SPA-SAT, outlined below and illustrated in FIG. 4, returns false if the set of conditions is not satisfiable, otherwise it returns true:"*)(Column 7, lines 7-9).

As per claims 15, 35, and 53, Ghazal teaches a method where a negation of OP is represented by the operator OP', and where determining the satisfiability of the query includes: assigning conditions of the form (X OP Y+C) to a set S1; assuring condition of the form (X+Y OP C) to a set S2; assigning conditions of the form (X OP C) to a set S3 (i.e. *"The function SPA-SAT, outlined below and illustrated in FIG. 4, returns false if the set of conditions is not satisfiable, otherwise it returns true: Function SPA-SAT begin 1. Convert all conditions to <= comparisons only using the following transformation (block 405, expanded in FIG. 5):"*)(Column 7, lines 10-14); replacing each conditions in set S2 with two conditions in the form (Y OP -X+C) and (X OP -y+c) (i.e. *"Convert (X<Y+C) to (X<=Y+(C-1)) (block 505)"*)(Column 7, line 2); if -X is present in set S2: for each condition in set S3: adding a condition of the form (-X OP' -C) to set S3 (i.e. *"Convert (X=C) to (X<=0+C) and (0<=X+(-C)) (block 545)"*)(Column 7, lines 24-25); and determining the satisfiability of the group of conditions (S1 UNION S2 UNION S3) (i.e. *"CREATE VIEW ORDERTBL AS SELECT * FROM FirstQOrders UNION SELECT * FROM*

SecondQOrders UNION SELECT * FROM ThirdQOrders UNION SELECT * FROM

FourthQOrders;")(Column 6, lines 38-42).

As per claim 19, Ghazal teaches a method for analyzing a query (i.e. *"The method*

includes analyzing the satisfiability of the conditions before executing the query.")(Column 1, lines 31-33),

the query including one or more conditions and one or more sub-queries, (i.e. *"The DBS*

100 usually receives queries and commands to build tables in a standard format, such as SQL.")(Column

5, lines 26-28), the conditions including one or more connecting conditions that introduce

the sub-query in the query, each of the sub-queries including zero or more conditions,

the method including (i.e. *"The conditions include one or more variables, zero or more SQL IN (inlist)*

conditions, and zero or more SQL NOT IN (not in list) conditions.")(Column 3, lines 50-53): creating a

Global Conditions set including one or more conditions representing one or more

connecting conditions (i.e. *"In general, in another aspect, the invention features a method for*

determining the satisfiability of and creating transitive closure in conditions in a database query. The

conditions include one or more variables, zero or more SQL IN (inlist) conditions, and zero or more SQL

NOT IN (not in list) conditions. The method includes for each variable in the query, defining an in list from

a SQL IN statement associated with the variable, a list from the SQL NOT IN statements not in list and

any other query statements that relate the variable to a constant with a operator, and an interval for the

range of values associated with the variable.")(Column 3, lines 48-61).

As per claim 20, Ghazal teaches a method for analyzing a query (i.e. *"The method*

includes analyzing the satisfiability of the conditions before executing the query.")(Column 1, lines 31-33),

the query including one or more conditions and one or more sub-queries (i.e. *"The DBS*

100 usually receives queries and commands to build tables in a standard format, such as SQL.")(Column

5, lines 26-28), the conditions including one or more connecting conditions that introduce

the sub-query in the query, each of the sub-queries including zero or more conditions,

the method including (i.e. *"The conditions include one or more variables, zero or more SQL IN (inlist)*

conditions, and zero or more SQL NOT IN (not in list) conditions.")(Column 3, lines 50-53): creating a

Transitive Closure set of conditions based on one or more connecting conditions (i.e. *"In*

general, in another aspect, the invention features a method for determining the satisfiability of and

creating transitive closure in conditions in a database query. The conditions include one or more

variables, zero or more SQL IN (inlist) conditions, and zero or more SQL NOT IN (not in list) conditions.

The method includes for each variable in the query, defining an in list from a SQL IN statement

associated with the variable, a list from the SQL NOT IN statements not in list and any other

querystatements that relate the variable to a constant with a operator, and an interval for the range of

values associated with the variable.")(Column 3, lines 48-61).

Contact Information

Any inquiry concerning this communication or earlier communications from the

examiner should be directed to Farhan M. Syed whose telephone number is 571-272-

7191. The examiner can normally be reached on 8:30AM-5:00 PM.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's

supervisor, Jeffrey Gaffin can be reached on 571-272-4146. The fax phone number for

the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the

Patent Application Information Retrieval (PAIR) system. Status information for

published applications may be obtained from either Private PAIR or Public PAIR.

Status information for unpublished applications is available through Private PAIR only.

For more information about the PAIR system, see http://pair-direct.uspto.gov. Should

you have questions on access to the Private PAIR system, contact the Electronic

Business Center (EBC) at 866-217-9197 (toll-free).

FMS

HOSAIN ALAM
SUPERVISORY PATENT EXAMINER